AMENDMENTS TO THE CLAIMS:

Please amend the following claims as follows:

1) (Currently Amended)   A system for recognizing a particular pattern formed by a plurality of bytes in a stream of bytes, said system operating in a series of states, and said system including:

>a plurality of ~~look-up~~ lookup tables, one for each of said plurality of bytes, each ~~look -up~~ lookup table having a section associated with each state~~,~~ ;

>a first index table which has index data associated with each state, the index data associated with each state comprising a plurality of index bits~~,~~ ; and

>a next state table which has a data word associated with each state, each data word having data that specifies the next state and/or a special flag,

said system performing the following steps in each particular state of a series of states~~,~~ :

>a) interrogating a particular section of each lookup table utilizing the value of the associated bytes to determine the value of a series of bits, said particular section of each lookup table being determined by the particular state~~,~~ ;

>b) generating a pointer from a combination of the value of the series of bits determined by said lookup tables and the bits in current state location in said first index table, said pointer specifying the address of a particular data word in said next state table~~,~~ ;

>c) reading the particular data word specified by the pointer generated in step "b" and based on the contents of said data word proceeding to the next state specified by said data word and/or performing special operations in response to said flag~~,~~ ; and

>d) repeating steps "a", "b" and "c" until an end of operation is indicated by said flag.

2) (Currently Amended)   The system recited in ~~table~~ claim 1 wherein all of said plurality of bytes are used to interrogate said ~~look-up~~ lookup tables in parallel.

3) (Original) The system recited in claim 1 wherein those bytes that cause the same transition are grouped into classes whereby multiple bytes can be represented by a single class, each class having a single entry in any particular single section of said table.

4) (Currently Amended) The system recited in claim 1 wherein at each state, each ~~look-up~~ lookup table generates a series of bits, the particular series of bits generated being dependent upon the particular value of the associated ~~of~~ byte.

5) (Currently Amended) The system recited in claim 4 wherein said ~~look-up~~ lookup tables generate a variable number of bits depending upon the particular state of operation.

6) (Currently Amended) The system recited in claim 3 wherein the bits generated by any ~~look-up~~ lookup table depends upon the number of character classes of the associated byte.

7) (Currently Amended) The system recited in claim 4 wherein the bits generated by said plurality of ~~look-up~~ lookup tables are concatenated together and then concatenated with said value from said first index table.

8) (Currently Amended) The system recited in claim 4 wherein the bits generated by said plurality of ~~look-up~~ lookup tables and said value from said first index table are combined to form the address of a location ~~is~~ in said next state table.

9) (Original) The system recited in claim 1 wherein said data stream comprises a stream of bytes that form a data packet, said particular operation is the recognition of a particular pattern of bytes in a data packet and wherein a signal is given when said pattern is recognized.

10) (Currently Amended) The system recited in claim 9 wherein when a particular pattern of bytes is recognized, said operation continues to another state to find another occurrence of said pattern of ~~byes~~ bytes in said packet.

11) (Currently Amended)   The method of recognizing a particular pattern formed by a
plurality of bytes in stream of bytes, said method operating in a series of states, and utilizing:

a plurality of ~~look-up~~ lookup tables, one for each of said plurality of bytes, each ~~look~~
~~up~~ lookup table having a section associated with each state~~,~~ :

a first index table for storing a plurality of index bits associated with each particular
state~~,~~ : and

a next state table for storing data words, each of which indicates the next state and/or
a special operation flag,

said method including the following steps in each particular state in a series of states~~,~~ :

a) interrogating a particular section of each lookup table utilizing the value of the
associated byte to determine the value of a series of bits, said particular section
of each ~~look-up~~ lookup table being based on the particular state~~,~~ :

b) generating a pointer which specifies the address of a data word in said next state
table from a combination of the selected values from the lookup tables and a
value from the first index table associated with the particular state~~,~~ :

c) reading the particular data work specified in step "b" and based on the contents of
said data word proceeding to the next state and/or performing operations
indicated by a flag in said data work~~,~~ : and

d) repeating steps "a", "b" and "c" until a flag in said next state table indicates an end
of the operation.

12) (Currently Amended)   The method recited in claim 11 wherein all of said plurality of
bytes are matched with data in said ~~look-up~~ lookup tables in parallel.

13) (Original)   The method recited in claim 11 wherein those bytes that cause the same
transition are grouped into classes whereby multiple bytes can be represented by a single
class, each class having a single entry in a section of said table.

14) (Currently Amended)   The method recited in claim 11 wherein at each state, each ~~look~~
~~up~~ lookup table generates a series of bits, the particular series of bits generated being

dependent upon the particular value of the associated byte.

15) (Original) The method recited in claim 14 wherein said tables generate a variable number of bits depending upon the particular state of operation.

16) (Currently Amended) The method recited in claim 11 wherein the bits generated by any ~~look-up~~ lookup table depends upon the class of the associated byte.

17) (Currently Amended) The method recited in claim 14 wherein the bits generated by said plurality of ~~look-up~~ lookup tables are concatenated together and then concatenated with said value from said first index table.

18) (Original) The method recited in claim 11 wherein said classes are a compressed representation of the alphabet used in said state machine.

19) (Currently Amended) The system recited in claim 1 wherein the series of bits generated by each ~~look-up~~ lookup table has a number of bits that can vary from state to state.

20) (Currently Amended) The method recited in claim 11 wherein the series of bits generated by each ~~look-up~~ lookup table has a number of bits that can vary from state to state.

21) (Currently Amended) The system recited in claim 1 wherein said system includes four ~~look-up~~ lookup tables.

22) (Original) The system recited in claim 1 wherein each of said tables has 256 entries for each state.

23) (Currently Amended) The system recited in claim 1 wherein said system includes a number of ~~look-up~~ lookup ~~table~~ tables which is a multiple of four.

24) (Currently Amended) The system recited in claim 1 wherein each of said ~~look-up~~ lookup tables has a number of entries equal to the values in the alphabet used by said bytes.

25) (Currently Amended) An integrated circuit for recognizing a plurality of characters in parallel, said integrated circuit operating in series of states, and said integrated circuit including~~:~~

      a plurality of ~~look-up~~ lookup tables, one for each of said characters, each ~~look-up~~ lookup table having a section associated with each state~~,~~ ;

      an index table for storing a plurality of index bits for each state~~,~~ ; and

      a next state table for storing data words, each of which indicates the next state of said state machine,

said integrated circuit performing the following steps in a plurality of particular states~~,~~ :

      a) interrogating the section of said ~~look-up~~ lookup table associated with said particular state and generating a series of bits based on the value of the associated character~~,~~ ;

      b) generating a pointer which specifies the address of a data word in said next state table from a combination of the selected values from the lookup tables and a value from said first index table~~,~~ ;

      c) reading the particular data word specified in step "b" and proceeding to the state specified by said data word~~,~~ ; and

      d) repeating steps "a", "b" and "c" until said operation ends.

26) (Currently Amended) The integrated circuit recited in claim 25 wherein the number of bits generated by said ~~look-up~~ lookup tables and the number of bits in the value from said current state table vary by state, and the total number of bits is the same in all states.

27) (Currently Amended) The integrated circuit recited in claim 25 wherein there are four ~~look-up~~ lookup tables and four characters are evaluated in parallel.

28) (Currently Amended) A method for evaluating in parallel a plurality of characters in a stream of bytes,

said method operating in a series of states and utilizing a set of alphabet ~~look up~~
<u>lookup</u> tables, one lookup table for each of said plurality of characters, each ~~look up~~
<u>lookup</u> table having a section associated with each of said states, and a next state
table, said method comprising:

a) selecting an alphabet lookup table from said set of alphabet tables, as a function of
a current state and byte position within said set of bytes, and obtaining a class
code from said selected alphabet table corresponding to the character
represented by said byte;

b) selecting a value from said next state table as a function of the current state, and
generating an indication of the next state from said class codes of said set of
bytes and from said value from said next state table~~;~~ <u>; and</u>

c) repeating steps "a" and "b" until said state machine arrives at acceptance state or
until entering a failure state.

29) (Currently Amended) A system for recognizing a particular pattern formed by a
number of bytes in a stream of bytes, said number being at least one, said
system operating in a series of states, and said system including<u>:</u>

a ~~look up~~ <u>lookup</u> table for each byte in said number of bytes, each ~~look up~~
<u>lookup</u> table having a section associated with each state~~;~~ <u>;</u>

a first index table which has index data associated with each state, the index
data associated with each state comprising a plurality of index bits~~;~~ <u>;</u>
and

a next state table which has a data word associated with each state, each data
word having data that specifies the next state and/or a special flag~~;~~ <u>;</u>

said system performing the following steps in each particular state of a series of states~~;~~ <u>:</u>

a) interrogating a particular section of each lookup table utilizing the value of
each of said bytes to determine the value of a series of bits, said
particular section of each lookup table being determined by the
particular state~~;~~ <u>;</u>

b) generating a pointer from a combination of the series of bits generated by

the lookup tables and bits in the current state location in said first index table, said pointer specifying the address of a particular data word in said next state table; :

c) reading the particular data word specified by the pointer generated in step "b" and based on the contents of said data word proceeding to the next state specified by said data word and/or performing special operations in response to said flag; : and

d) repeating steps "a", "b" and "c" until an end of operation is indicated by said flag.

30) (Original) The system recited in claim 1 wherein said system includes a mask with one bit for each of said plurality of bytes in a data packet, and said steps including an initial step of selecting bits for processing utilizing said mask.

31) (Original) The method recited in claim 10 wherein said method utilizes a mask with one bit for each of said plurality of bytes in a data packet, and said steps including an initial step of selecting bits for processing utilizing said mask.